Desenvolvimento de um Modelo Exato de Alocação de Disciplinas no Contexto de um Curso de Graduação

Rudson Francisco da Silva Mendes, Fernando Concatto, Rafael de Santiago
Universidade do Vale do Itajaí
R. Uruguai, 458 - Centro, 88302-202 – Itajaí – SC
rudsonm@edu.univali.br, fernandoconcatto@gmail.com, rsantiago@univali.br

RESUMO

O problema de agendamento no âmbito educacional é chamado de *University Timetabling*, cujo objetivo é alocar disciplinas em determinados dias e horários ao longo de uma semana, de forma que esta se repita por todo o período letivo. Porém, uma série de restrições precisam ser estabelecidas para realizar estas alocações. Sendo de dois tipos, estas restrições se caracterizam entre fortes e fracas, sendo que as fortes não podem ser violadas de maneira alguma e as fracas devem ser satisfeitas o máximo possível. O trabalho reportado neste artigo tem como objetivo desenvolver um modelo exato de programação inteira para solucionar o problema de agendamento do quadro de horários de disciplinas do curso de graduação Ciência da Computação, onde as restrições fortes e fracas levaram em consideração características específicas da realidade do curso. A grade resultante obtida pelo modelo obteve a mesma qualidade da solução atualmente utilizada, que é obtida de maneira analítica pela instituição de ensino. O novo procedimento automatizado proposto neste artigo resultou em uma aceleração expressiva em relação ao tempo necessário para obtenção da solução analítica, segundo informações fornecidas pelo coordenador do curso.

PALAVRAS CHAVE. Programação Inteira, Timetabling, Universidade.

Programação Matemática, Otimização Combinatória

ABSTRACT

The problem of scheduling in the educational context is called University Timetabling, whose objective is to allocate lectures in timeslots of a week in a manner that can be repeated during the school period. A set of constraints must be established to accomplish the timetabling. Those constraints might be either hard or soft, where a hard restriction indicates that is cannot be violated in any circumstance, and soft constraints should be satisfied as much as possible. This paper aims to develop an integer linear programming model to solve the problem of constructing a timetable for the classes of a Bachelor in Computer Science course, where the hard and soft constraints took into account specific characteristics of the course's context. The timetabling derived from the model obtained a quality equivalent to the actual analytical solution utilized by the target educational institution; however, compared to the currently employed manual procedure, our approach solved the problem approximately 14 million times faster.

KEYWORDS. Integer Programming. Timetabling. University.

Mathematical Programming, Combinatorial Optimization

https://proceedings.science/p/85466?lang=pt-br

1. Introdução

O problema de programação de tabelas de horários (*timetabling*) é recorrente no ambiente de instituições educacionais. A cada início de um período letivo de uma instituição são construídas grades de horários de disciplinas, sendo que este processo basicamente consiste em alocar disciplinas em determinados dias e horários. A alocação de disciplinas é essencial para o correto funcionamento de uma instituição de ensino, dado que é este o processo que especifica os dias e horários em que as aulas ocorrerão [Even et al., 1975; Babaei et al., 2015; Johnes, 2015].

Há uma grande dificuldade em encontrar soluções satisfatórias para este tipo de problema, o que tem motivado diversas pesquisas e até ferramentas que auxiliam neste processo [Babaei et al., 2015]. A construção de uma grade de horários, por vezes, é vista como uma tarefa simples, graças ao grande número de possíveis combinações ou soluções para o problema. Entretanto, fatores como indisponibilidade de professores em determinados dias e horários e conflitos entre disciplinas lecionadas pelo mesmo professor limitam significativamente o número de possíveis soluções, o que acaba tornando a resolução deste problema uma tarefa extremamente difícil de ser resolvida de maneira analítica. Além disso, há a possibilidade de novos professores serem admitidos para assumir o cargo de professores que se desligaram da instituição ou foram migrados para outras disciplinas, implicando na alteração das indisponibilidades de horários, fazendo assim necessária a mudança ou até mesmo a construção de uma nova grade de horários [Babaei et al., 2015; Silva e Campos, 2017].

Para o trabalho reportado neste artigo é utilizado um modelo de programação inteira baseado nas particularidades do problema de agendamento de alocação de disciplinas nos dias da semana. É então utilizada uma função objetivo para avaliar soluções, deste modo é possível determinar critérios decisórios na busca pela melhor solução para o problema. Neste modelo, foram impostas algumas restrições, sendo estas classificadas como fortes ou fracas. Soluções que não atendem às restrições fortes são definidas como inadmissíveis, portanto não apresentam utilidade como solução para o problema e são descartadas. As restrições do tipo fracas contribuem na função objetivo apenas penalizando a qualidade de soluções. Os dados utilizados durante os experimentos foram obtidos com a coordenação do curso de Ciência da Computação da Universidade do Vale do Itajaí para o semestre 2018/1.

Este artigo se divide em mais cinco seções. Primeiramente, se apresenta uma breve fundamentação teórica sobre o tema. Na seção seguinte, é apresentada a atual realidade do curso utilizado como base para o desenvolvimento deste trabalho, sendo este o curso de Bacharelado em Ciência da Computação na instituição de ensino Universidade do Vale do Itajaí, campus Itajaí. Na sequência, o modelo proposto é apresentado detalhadamente, e os resultados obtidos são demonstrados e discutidos. Quanto aos resultados, é apresentada a grade de horários resultante da solução exata obtida através do modelo proposto, sendo subsequentemente comparada com a solução analítica utilizada atualmente. Ao final, considerações são apresentadas e trabalhos futuros são sugeridos.

2. Fundamentação Teórica

Instituições acadêmicas frequentemente enfrentam ou já enfrentaram problemas de *timetabling* para situações como programação de horários de aulas, eventos, bancas de avaliação de trabalhos acadêmicos, entre outros. O desenvolvimento de uma ferramenta para resolução de problemas deste tipo não é tão simples quanto pode aparentar, devido à sua natureza combinatória e complexidade de ordem NP-difícil, um aspecto provado por [Even et al., 1975]. Além disso, estratégias de modelagem do problema em diferentes universidades podem apresentar um grande número de particularidades, pois cada instituição possui características específicas.

O problema *timetabling* no âmbito acadêmico é chamado de *university course timetabling*. Ele possui característica combinatória e consiste em agendar aulas de cursos em um conjunto de salas e horários disponíveis. A tentativa de solucionar este problema de forma analítica requer uma quantidade considerável de tempo; dependendo do tamanho do problema, pode tomar de algumas

https://proceedings.science/p/85466?lang=pt-br

horas a até semanas. Entretanto, esta dificuldade de solucionar o problema tem sido motivo de mobilização de diversas pesquisas, a fim de propôr técnicas e ferramentas para solução do *university timetabling problem*. Estas constribuições vêm principalmente de campos como pesquisa operacional [Johnes, 2015], utilizando recursos de algoritmos, programação inteira, heurísticas e meta-heurísticas.

A natureza combinatória do problema restringe consideravelmente a quantidade de agendamentos a serem realizados em um *university course timetabling*. O número de salas disponíveis para aula, o número de horários livres ao longo de um dia e semana e a indisponibilidade de professores para alguns horários e dias são alguns exemplos de fatores que dificultam obter computacionalmente uma solução em tempo viável. Para se adequar ao modelo do mundo real e a estas limitações, algumas restrições devem ser estabelecidas, e estas podem ser classificadas em dois tipos: *(i)* restrições fortes, que são condições que essencialmente devem ser satisfeitas, caso contrário uma solução que não satisfaça estas condições é definida inadmissível, e *(ii)* restrições fracas, as quais podem ser violadas, porém devem ser respeitadas o máximo possível, caso contrário são penalizadas.

No trabalho de Babaei et al. [2015], uma revisão da literatura é apresentada a respeito do *timetabling problem*. Os autores descrevem termos e significados comuns a trabalhos que abordam o mesmo problema e um conjunto de restrições fortes essenciais para que soluções admissíveis sejam aplicáveis ao mundo real. Sugestões para restrições fracas e uma grande variedade de abordagens para a resolução do problema são sugeridas, incluindo coloração de grafos, programação inteira, meta-heurísticas e sistemas multiagentes. Os autores recomendam que estratégias exatas(como programação inteira) sejam utilizadas somente quando o tamanho do problema for relativamente pequeno, pois torna-se necessário utilizar abordagens heurísticas em problemas maiores devido à natureza NP-difícil do problema.

Uma abordagem de resolução do *university timetabling problem* por meio de programação inteira foi descrita por Daskalaki et al. [2004], onde o contexto sob estudo consiste em universidades da Grécia. O modelo proposto pelos autores leva em consideração diversos aspectos particulares do país em questão; entretanto, o mesmo apresenta alta flexibilidade e abrangência, podendo ser facilmente adaptado para outros contextos, como o deste trabalho. O programa foi solucionado por meio da ferramenta CPLEX 5.1, e segundo os autores, forneceu soluções válidas e aplicáveis aos departamentos da universidade estudada.

Para resolver o problema *university course timetabling*, Silva e Campos [2017] utilizou uma matriz *X* de quatro dimensões em seu modelo. Para cada dimensão é atribuído um conjunto *P* de professores, *D* de disciplinas, *H* de horários e *J* de dias. Portanto, o professor *p* leciona a disciplina *d* no horário *i* no dia *j*, caso a célula X_{pdij} possuir 1 como valor. É utilizado também no modelo de Silva e Campos [2017] uma matriz H_{pd} que identifica se o professor *p* está apto a lecionar a disciplina *d*, implicando que a designação de um professor para uma disciplina não ocorre previamente e sim durante a solução do problema.

Lü e Hao [2010] propuseram uma busca tabu adaptativa para solucionar o problema. A solução proposta consiste em três fases: inicialização, intensificação e diversificação. A etapa de inicialização trata de construir uma solução inicial, que consiste em uma programação de horário aleatória que satisfaça todas as restrições fortes impostas. Após obter uma solução inicial, os autores utilizam uma abordagem adaptativa entre intensificação e diversificação do espaço de busca, a fim de reduzir o número de violações de restrições fracas, porém sempre satisfazendo o conjunto de restrições fortes. Assim, sua abordagem tende a uma melhora na qualidade das soluções sem arriscar a admissibilidade das mesmas.

Em seu modelo, Feng et al. [2017] possuem restrições fortes e fracas que refletem a necessidade de programação de horários na maioria das universidades da China e Coreia do Sul. Disciplinas que possuem aulas de múltiplos horários podem ter seus agendamentos baseados em periodicidade ou consecutividade. Caso agendadas baseadas em periodicidade, as aulas acontecem

https://proceedings.science/p/85466?lang=pt-br

em apenas um determinado período, não podendo ocorrer em dias sequenciais; quando baseadas em consecutividade, são agendadas de forma semanal, com aulas se repetindo ao longo do semestre em uma mesma sala. Além de agendar aulas em horários, os autores preocupam-se em alocar estas aulas em salas também, levando em conta a ocupação de alunos de uma determinada disciplina para a capacidade que uma sala comporta. Portanto, quanto maior a diferença de alunos de uma disciplina para a quantidade que uma sala comporta, maior será a penalização desta alocação.

3. Realidade do Curso

O Curso de Ciência da Computação ofertado pela Universidade do Vale do Itajaí acontece somente no período noturno. O curso é composto por um total de 10 semestres, onde estes possuem uma série de disciplinas com carga horária suficiente para preencher uma semana de dias letivos. Uma semana é composta por cinco dias letivos, onde para cada um destes dias há dois horários de aula que totalizam 3 horas e meia. Cada um dos semestres possuem disciplinas suficientes para preencher todos os horários ao longo de uma semana.

A construção de uma grade de horário para um curso basicamente consiste em alocar disciplinas para preencher horários. Um grande limitador ou dificuldade de construir uma grade de horários é a disponibilidade dos professores ou até mesmo as preferências de horários que os mesmos possuem ao longo da semana. Na instituição alvo deste trabalho no semestre de 2018/1 os professores possuem disponibilidade semanal em média de 65% (como pode ser verificado na Tabela 3). Quanto menor a média de disponibilidade dos professores maior se torna a dificuldade de encontrar uma grade de horários válida.

Atualmente as programações de horário são feitas de maneira analítica, onde deve ser construída uma programação para cada curso. O responsável por gerar e montar esta programação geralmente é o coordenador do mesmo curso para qual a programação está sendo destinada. Por meio de uma entrevista com o coordenador do curso de graduação de Ciência da Computação, foi constatado que o processo de construção de uma programação de horários para o curso leva em torno de 4 horas. Além disso, os dados necessários para gerar a programação acadêmica, como a disponibilidade dos professores envolvidos ao longo de uma semana, são coletados de forma manual, onde cada professor deve preencher um formulário e entregá-lo ao coordenador do curso.

Diferentemente do contexto acadêmico em que Silva e Campos [2017] desenvolveu seu modelo, onde os professores possuem disciplinas que estão aptos a lecionar e somente no momento da construção da grade é que são devidamente designados, na instituição alvo deste trabalho as disciplinas que compõem a grade curricular de um curso são designadas a serem lecionadas por professores antes mesmo do período letivo de um semestre iniciar. Portanto, a alocação de uma disciplina m na segunda-feira seria equivalente a alocar o professor p que ministra a aula m na segunda-feira.

Antes do período letivo de cada semestre iniciar, todo aluno deve montar sua programação acadêmica. Este processo consiste em um aluno preencher horários ao longo de uma semana, selecionando disciplinas que estão aptas a serem cursadas pelo mesmo. Devido a necessidade de requerer previamente o conhecimento de uma certa gama de assuntos, há disciplinas que para serem cursadas necessitam que outras disciplinas sejam concluídas(pré-requisitos). Devido a impossibilidade de disciplinas deste tipo serem cursadas simultâneamente por um aluno, é preferível que uma disciplina e outras que são suas pré-requisitos aconteçam num mesmo dia da semana. Este fator facilita na seleção da programação acadêmica dos alunos do curso, justamente por possuir mais opções de disciplinas aptas a serem cursadas em dias distintos.

4. Modelo Proposto

Para resolver o problema proposto neste trabalho, foi desenvolvido um modelo exato de programação inteira. Este modelo possui alguns parâmetros, que devem ser informadas previamente a execução do modelo, geralmente utilizados somente para leitura, é através destes parâmetros que as variáveis de decisão(de tempo de execução) são definidas. Os parâmetros quantitativos de peso

https://proceedings.science/p/85466?lang=pt-br

P_{dist}, $P_{evitaveis}$ e P_{eqdias} definem as penalidades do modelo para as restrições fracas, sendo estas expressas como termos da função objetivo a ser minimizada, enquanto as demais são utilizadas tanto para a definição das restrições fortes quanto como auxiliares na especificação da função objetivo.

- S: o conjunto de semestres de um curso;
- D: o conjunto de dias letivos da semana;
- H: o conjunto de horários de cada dia;
- M: o conjunto de disciplinas do curso;
- F: o conjunto de professores do curso;
- R: conjunto de pares de dias e horários em que determinado professor f não está disponível para lecionar;
- E: conjunto de pares de dias e horários em que determinado professor f está disponível, porém não deseja ou prefere evitar lecionar;
- C: o conjunto de cargas horárias para cada disciplina d;
- S: conjunto que define o semestre no qual uma disciplina m se encontra;
- P_{dist}: peso quantitativo para uma mesma disciplina alocada em horários distantes;
- $P_{evitaveis}$: peso para alocações de professores em horários não desejados pelos mesmos;
- P_{eqdias}: peso para as alocações de disciplinas que ocorrem num mesmo dia que suas respectivas disciplinas de pré-requisito.

A solução derivada do modelo é apresentada como uma matriz T de quatro dimensões. O valor contido em T_{sdhm} é binário, onde o valor 1 é interpretado como uma alocação da disciplina m do semestre s que está no dia d e no horário h e 0 indica a ausência da mesma. Na aplicação do modelo, trabalhou-se com os horários reais de 19h00–20h40 para o primeiro horário e 20h50–22h30 para o segundo horário, e levou-se em conta todas as 47 disciplinas ofertadas pelo curso.

O objetivo do modelo proposto é minimizar a ocorrência de um conjunto de restrições fracas; ou seja, quanto menos violações para este conjunto de restrições fracas uma solução possuir, melhor será sua avaliação. Além disso, o modelo apresenta algumas restrições fortes que caso restringidas, desqualificam a solução, tornando-a inadmissível para o problema, como por exemplo soluções que alocam uma sala para duas disciplinas distintas no mesmo dia e horário. A seguir, a função objetivo e as restrições do modelo são definidas.

$$\begin{aligned}\textbf{minimizar } \Big\{ & \sum_{d\in D}\sum_{m\in M} P_{dist} X_{S_m dm} + \sum_{f\in F}\sum_{m\in M_f}\sum_{(d,h)\in E} P_{evitaveis} T_{S_m dhm} + \\ & \sum_{d\in D}\sum_{m_a\in\{1,\ldots,M\}}\sum_{mb\in\{m_a+1,\ldots,M\}} P_{eqdias}\big(4 - T_{S_{m_a} d0 m_a} - T_{S_{m_a} d1 m_a} \\ & - T_{S_{m_a} d0 m_b} - T_{S_{m_a} d1 m_b}\big)\Big\} \end{aligned} \tag{1a}$$

https://proceedings.science/p/85466?lang=pt-br

sujeito a:

$$\sum_{s\in S}\sum_{d\in D}\sum_{h\in H} T_{sdhm} = C_m, \qquad \forall m \in M \qquad \text{(2a)}$$

$$\sum_{s\in S\setminus\{S_m\}}\sum_{d\in D}\sum_{h\in H} T_{sdhm} = 0, \qquad \forall m \in M \qquad \text{(2b)}$$

$$\sum_{m\in M} T_{S_m dhm} \leq 1, \qquad \forall s \in S, d \in D, h \in H \qquad \text{(2c)}$$

$$T_{S_{m_a} dhm_a} T_{S_{m_b} dhm_b} \leq 1, \qquad \forall d \in D, \forall h \in H, \forall f \in F, \forall m_a \in M_f, \forall m_b \in M_f \setminus \{m_a\} \qquad \text{(2d)}$$

$$\sum_{m\in M_f} T_{S_m dhm} = 0, \qquad \forall f \in F, (d,h) \in R_f \qquad \text{(2e)}$$

$$X_{S_m dm} \geq T_{S_m d0m} - T_{S_m d1m}, \qquad \forall d \in D, \forall m \in M \qquad \text{(2f)}$$

$$X_{S_m dm} \geq T_{S_m d1m} - T_{S_m d0m}, \qquad \forall d \in D, \forall m \in M \qquad \text{(2g)}$$

$$T_{sdhm} \in \{0,1\}, \qquad \forall s \in S, \forall h \in H, \forall d \in D, \forall m \in M \qquad \text{(2h)}$$

$$X_{sdm} \in \{0,1\}, \qquad \forall s \in S, \forall d \in D, \forall m \in M. \qquad \text{(2i)}$$

No modelo desenvolvido neste trabalho, um conjunto de restrições fortes precisa ser estabelecido a fim de se adequar ao contexto da universidade alvo. As restrições deste conjunto são as seguintes:

- **2a**: garante que determinada disciplina *m* será alocada em uma quantidade equivalente as alocações necessárias em C_m;

- **2b**: define que para todas as disciplinas em *M* só poderá haver alocações em seus respectivos semestres, portanto uma disciplina *m* só pode ser alocada no semestre S_m;

- **2c**: garante que dentre o conjunto de disciplinas *M* só poderá haver nenhuma ou uma alocação de uma única disciplina *m* em determinado semestre, dia e horário;

- **2d**: não permite que qualquer professor ministre duas ou mais aulas num mesmo dia e horário; ou seja, o professor não poderá estar presente em dois ou mais lugares ao mesmo tempo;

- **2e**: é utilizado um conjunto *R* de pares de dias e horários em que determinado professor *f* não está disponível para lecionar, impossibilitando a ocorrência de alocações de disciplinas ministradas por *f* em dias e horários R_f que este não está disponível.

- **2f / 2g**: de acordo com estas duas restrições, é possível certificar de que em uma célula X_{sdm} é admitido o valor 1 caso a disciplina *m* esteja alocada nos dois horários do dia *d* e 0 caso contrário. Por exemplo, supondo que para uma mesma disciplina *m*, $T_{S_m d0m}$ seja 1 e $T_{S_m d1m}$ seja 0, a subtração em **2f** resulta 1 e em **2g** resulta -1, portanto de acordo com a limitação superior de **2g**, em $X_{S_m dm}$ é admitido o valor 1. De acordo com a função objetivo do modelo, há uma penalização de *m* por não possuir alocações sequenciais no dia *d*. Entretanto, para alocações sequenciais de *m* em um mesmo dia *d*, $T_{S_m d0m}$ e $T_{S_m d1m}$ são 1, é então obtido 0 para $X_{S_m dm}$ de acordo com a subtração nas duas restrições, não resultando em penalização para esta alocação.

https://proceedings.science/p/85466?lang=pt-br

Tabela 1: Ssemestre, carga horária, professor e requisitos considerados para cara disciplina.

Id.	Disciplina	Sem.	C.H.	Prof.	Req.	Id.	Disciplina	Sem.	C.H.	Prof.	Req.
1	Intro C.Comp.	1	1	1	–	25	Sis.Operac.	5	2	13	21
2	Mat.Comp.	1	1	1	–	26	Tóp.Esp.Prog.	5	2	6	19
3	Álgebra	1	2	7	–	27	Banco de Dados II	6	2	16	24
4	Alg.Prog. I	1	4	20	–	28	Eng.Soft. II	6	2	2	22
5	Trab.Acad.	1	1	17	–	29	Simulação	6	2	3	15
6	Soc.Ética, MA	1	1	11	–	30	Grafos	6	2	25	–
7	Cálculo I	2	2	7	–	31	Ling.Formais	6	2	24	–
8	Inglês	2	2	17	–	32	Redes I	7	2	13	–
9	Metodologia	2	1	17	–	33	IA I	7	2	26	–
10	Circ.Digitais	2	2	1	2	34	Complexidade	7	1	24	–
11	Alg.Prog. II	2	3	14	4	35	Compiladores	7	2	9	31
12	Cálculo II	3	2	7	7	36	Eng.Soft. III	7	3	5	28
13	Ética	3	1	0	–	37	Adm.	8	2	28	–
14	Arq.Org.Comp. I	3	2	8	10	38	Redes II	8	2	13	32
15	Est.Dados	3	3	24	11	39	Sis.Distrib.	8	2	9	–
16	Prog. I	3	2	6	–	40	IA II	8	2	22	33
17	Cálc.Num.	4	2	15	12	41	Inf.Educação	8	2	4	–
18	Model.Incert.	4	2	15	–	42	Tóp.Esp.Comp. I	9	2	25	–
19	Prog. II	4	3	10	–	43	TTC I	9	4	–	–
20	Inf.Soc.	4	1	0	–	44	Empreendedorismo	9	2	19	–
21	Arq.Org.Comp. II	4	2	13	14	45	Usabilidade	9	2	12	–
22	Eng.Soft. I	5	2	2	–	46	TTC 2	10	8	–	–
23	Estatística	5	2	18	–	47	Tóp.Esp.Comp II	10	2	21	–
24	Banco de Dados I	5	2	16	–						

4.1. Detalhamento de Restrições

O curso possui disciplinas com carga horária distintas, onde a variação de ocupação semanal necessária é de um a quatro horários. A primeira restrição fraca está atrelada ao parâmetro P_{dist}, penalizando as alocações de uma mesma disciplina em dias distintos, favorecendo que as aulas de uma mesma disciplina ocorram num mesmo dia em horários sequenciais, aproveitando assim ao máximo a disponibilidade do professor para este dia. Uma matriz binária de três dimensões X é utilizada para armazenar as alocações não sequenciais de disciplinas. De acordo com as restrições fortes **2f** e **2g**, em determinada célula X_{sdm} é atribuído o valor 1 se a disciplina d do semestre s não esteja alocada sequencialmente nos dois horários do dia d, caso contrário seu valor será 0.

Há dias em que professores possuem compromissos que os incapacitam de lecionar em determinados dias e horários como também há casos em que determinados professores não desejam ou tendem a evitar lecionar em alguns dias. Para este segundo caso é definida uma restrição do tipo fraca, justamente por ser facultativa a disponibilidade dos professores, apenas tentando evitar alocá-los em determinados dias e horários não desejados. Juntamente com o parâmetro $P_{evitaveis}$ são penalizadas todas e quaisquer alocações do professor f que ocorram em qualquer um dos pares de dias e horários evitáveis pelo mesmo, os quais são definidos pelo conjunto E.

Como discutido na seção anterior, há disciplinas que para serem cursadas demandam a conclusão de um conjunto de outras disciplinas, sendo estas chamadas de disciplinas pré-requisito. Devido à indisponibilidade de cursar uma disciplina e qualquer uma de suas disciplinas pré-requisito simultaneamente, é preferível que estas ocorram em um mesmo dia da semana. Este aspecto auxilia os acadêmicos na construção de suas respectivas programações acadêmicas. P_{eqdias} é um peso quantitativo que impacta na penalização de alocações de disciplinas cuja aula ocorre em dias diferentes de suas disciplinas de pré-requisito.

5. Resultados

Esta seção apresenta os resultados obtidos pelo modelo exato de programação inteira desenvolvido no presente trabalho. A resolução do modelo foi obtida através da biblioteca do *solver* IBM ILOG CPLEX para a linguagem C++. O ambiente utilizado para experimento foi um

https://proceedings.science/p/85466?lang=pt-br

Tabela 2: Grade de horários para o curso de Ciência da Computação obtido pelo modelo proposto.

Sem.	**Horário**	2^a**-feira**	3^a**-feira**	4^a**-feira**	5^a**-feira**	6^a**-feira**
1	19:00–20:40	Alg.Prog. I	Mat.Comp.	Alg.Prog. I	Intro C.Comp.	Álgebra
	20:50–22:30	Alg.Prog. I	Trab.Acad.	Alg.Prog. I	Soc.Ética, MA	Álgebra
2	19:00–20:40	Inglês	Metodologia	Circ.Digitais	Cálculo I	Alg.Prog. II
	20:50–22:30	Inglês	Alg.Prog. II	Circ.Digitais	Cálculo I	Alg.Prog. II
3	19:00–20:40	Cálculo II	Est.Dados	Arq.Org.Comp. I	Prog. I	Est. Dados
	20:50–22:30	Cálculo II	Est.Dados	Arq.Org.Comp. I	Prog. I	Ética
4	19:00–20:40	Prog. II	Inf.Soc.	Arq.Org.Comp. II	Cálc.Num.	Model.Incert.
	20:50–22:30	Prog. II	Prog. II	Arq.Org.Comp. II	Cálc.Num.	Model.Incert.
5	19:00–20:40	Eng.Soft. I	Sis.Operac.	Tóp.Esp.Prog.	Banco de Dados I	Estatística
	20:50–22:30	Eng.Soft. I	Sis.Operac.	Tóp.Esp.Prog.	Banco de Dados I	Estatística
6	19:00–20:40	Ling.Formais	Eng.Soft. II	Banco de Dados II	Simulação	Grafos
	20:50–22:30	Ling.Formais	Eng.Soft. II	Banco de Dados II	Simulação	Grafos
7	19:00–20:40	Redes I	IA I	Compiladores	Complexidade	Eng.Soft. III
	20:50–22:30	Redes I	IA I	Compiladores	Eng.Soft. III	Eng.Soft. III
8	19:00–20:40	Sis.Distrib.	Inf.Educação	IA II	Adm.	Redes II
	20:50–22:30	Sis.Distrib.	Inf.Educação	IA II	Adm.	Redes II
9	19:00–20:40	TTC I	Usabilidade	Empreendedorismo	Tóp.Esp.Comp. I	TTC I
	20:50–22:30	TTC I	Usabilidade	Empreendedorismo	Tóp.Esp.Comp. I	TTC I
10	19:00–20:40	TTC 2	Tóp.Esp.Comp. II	TTC 2	TTC 2	TTC 2
	20:50–22:30	TTC 2	Tóp.Esp.Comp. II	TTC 2	TTC 2	TTC 2

computador com sistema operacional Linux, processador Intel i7-7500U, clock de 2.7GHz e 16GB de memória RAM.

Os dados utilizados para experimentação foram fornecidos pelo coordenador do curso de Ciência da Computação da instituição alvo deste presente trabalho. Como pode ser visto na Tabela 1, o problema é caracterizado por 10 semestres com um total de 47 disciplinas. Cada semestre possui um conjunto de disciplinas, 29 professores, um conjunto de horários em que os professores não estão disponíveis (equivalente a cerca de 35% do total de horários), e 5 dias que compõem uma semana cujo cada dia possui 2 horários disponíveis.

Para definição do valor dos parâmetros de peso utilizados para flexibilidade do modelo, um professor da instituição alvo foi consultado. Com o apoio deste especialista, os valores de 1000 para P_{dist}, 1 para $P_{evitaveis}$ e 1000 para P_{eqdias} foram definidos. Com estas configurações e no ambiente descrito previamente foi então obtida a solução exata apresentada na Tabela 2, em tempo computacional de 100ms, sendo que a mesma possui qualidade de 512,00.

Na tabela 3, são apresentados os dados dos dias em que os professores estão indisponíveis; estes dados são utilizados para constituir o conjunto *R*. Professores cuja disponibilidade de horários são baixas restringem consideravelmente o número de possíveis soluções para o problema. Quanto menor o nível de disponibilidade de um determinado professor, menores serão as opções de horários para alocar o mesmo. Este fator acaba por limitar a quantidade de possíveis combinações de alocações, portanto dificultando consideravelmente a resolução do problema. Porém quanto menor a disponibilidade destes professores maior será o número de soluções que se tornam inadmissíveis, prejudicando a possibilidade de obter uma solução de maior qualidade.

Para validar e verificar o modelo exato desenvolvido, é realizada uma comparação dos resultados obtidos por este modelo e a atual solução analítica que é utilizada pela instituição alvo deste trabalho. Foi verificado que a solução analítica possui a mesma qualidade da solução exata derivada do modelo. Porém há algumas disciplinas alocadas em horários diferentes entre as soluções comparadas, e mesmo com estas alocações divergentes não há qualquer diferença de qualidade, devido a sofrerem as mesmas penalidades.

Como previamente dito, foi relatado pelo coordenador do curso alvo que o tempo médio necessário para a construção da programação de horários de maneira analítica é de 4 horas. Logo, em relação a tempo, o modelo desenvolvido neste trabalho apresenta uma significante melhoria por

https://proceedings.science/p/85466?lang=pt-br

Tabela 3: Tabela de disponibilidade de professores. O símbolo • indica que o professor não pode lecionar naquele dia; um espaço em branco indica que o professor está disponível.

Prof.Id.	2^a**-feira**	3^a**-feira**	4^a**-feira**	5^a**-feira**	6^a**-feira**	**Prof.Id.**	2^a**-feira**	3^a**-feira**	4^a**-feira**	5^a**-feira**	6^a**-feira**
0						15	•	•	•		
1	•				•	16					•
2			•	•		17			•	•	•
3	•	•				18	•			•	
4			•		•	19		•		•	•
5			•			20		•		•	•
6		•			•	21			•	•	•
7		•	•			22	•	•		•	
8	•	•			•	23			•		•
9						24					
10					•	25					
11						26	•				•
12				•	•	27					
13						28	•	•	•		•
14											

ser parte de um processo computacional. Apesar de ser de natureza NP-difícil, a obtenção de 100ms para resolver o modelo se deve ao tamanho da instância, a qual é utilizada somente para um curso, o que a torna trivial.

6. Conclusão

Neste trabalho, foi proposto um modelo de programação linear para realizar a alocação de disciplinas em horários para o curso superior de Ciência da Computação da Universidade do Vale do Itajaí. São utilizados três tipos de restrições fracas, as quais são implementadas como penalizações na qualidade da solução, e foram projetadas restrições fortes através de restrições técnicas para o modelo, sendo que a violação de apenas uma restrição forte acaba por tornar a solução inadmissível. Mesmo a solução exata do modelo proposto apresentando qualidade equivalente a solução atualmente utilizada na instituição alvo, o tempo requerido para completar esta tarefa é significativamente menor por tratar-se de um processo computacional e não mais analítico.

A disponibilidade de professores em horários ao longo da semana influencia diretamente no número de soluções admissíveis para o problema. Há casos onde um baixo índice de disponibilidade impossibilita a existência de uma solução admissível para o problema; em casos como estes, um ponto a ser explorado é a alteração automática das disponibilidades ou até mesmo a troca de disciplinas entre professores, a fim de tornar uma solução admissível.

No âmbito educacional há alguns aspectos que o presente modelo não comporta. A implementação e modelagem destes aspectos seriam um ponto interessante para um próximo passo de desenvolvimento. Por exemplo, o modelo se limita a apenas um curso, mas poderia ser expandido para cursos de um mesmo centro, como o Centro de Ciências Tecnológicas da Terra e do Mar. É bastante comum o fato de professores lecionarem disciplinas de cursos distintos de um mesmo centro acadêmico. Um modelo mais abrangente se beneficiaria deste fato por reaproveitar informações dos professores que estes cursos possuem em comum. Além disso, seria necessário apenas uma execução do modelo, ao invés de uma vez para cada um dos cursos envolvidos; entretanto, isto implicaria na necessidade de uma maior robustez por parte do modelo.

Referências

Babaei, H., Karimpour, J., e Hadidi, A. (2015). A survey of approaches for university course timetabling problem. *Computers & Industrial Engineering*, 86:43–59. ISSN 0360-8352. URL `http://www.sciencedirect.com/science/article/pii/S0360835214003714`.

Daskalaki, S., Birbas, T., e Housos, E. (2004). An integer programming formulation for a case study in university timetabling. *European Journal of Operational Research*, 153(1):117–135.

https://proceedings.science/p/85466?lang=pt-br

Even, S., Itai, A., e Shamir, A. (1975). On the complexity of time table and multi-commodity flow problems. In *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)*, p. 184–193.

Feng, X., Lee, Y., e Moon, I. (2017). An integer program and a hybrid genetic algorithm for the university timetabling problem. *Optimization Methods and Software*, 32(3):625–649. ISSN 10294937. URL http://dx.doi.org/10.1080/10556788.2016.1233970.

Johnes, J. (2015). Operational research in education. *European Journal of Operational Research*, 243(3):683–696. ISSN 03772217. URL http://dx.doi.org/10.1016/j.ejor.2014.10.043.

Lü, Z. e Hao, J. K. (2010). Adaptive Tabu Search for course timetabling. *European Journal of Operational Research*, 200(1):235–244. ISSN 03772217. URL http://dx.doi.org/10.1016/j.ejor.2008.12.007.

Silva, I. L. e Campos, S. C. (2017). Programação inteira para timetabling em uma universidade privada.

https://proceedings.science/p/85466?lang=pt-br